Homework V

Due date: April 17, 2024 (11:59pm)

Problem 1: Support Vector Machine (35 points)

SVM has been designed with different formulations to tackle problems from various angles. For a dataset with N data points, denote $\mathbf{x}_i$ to the feature vector for the i-th data point, and y_i to be the label for the i-th data point. Consider the following formulation of SVM:

$$\begin{aligned} &\max_{\mathbf{w},b} \min_{1\le i\le N} \frac{\mathbf{w}^\top \mathbf{x}_i + b}{\|\mathbf{w}\|} \\ &s.t.\ y_1(\mathbf{w}^\top \mathbf{x}_1 + b) \ge 0, \\ &\quad \dots \\ &\quad y_N(\mathbf{w}^\top \mathbf{x}_N + b) \ge 0. \end{aligned} \tag{1}$$

Suppose we have a data set of two classes as in Figure 1. Each class has two data points, where each point $x_i \in \mathbb{R}^2$, with the first dimension of the feature denoted by the horizontal axis and the second by the vertical axis.

Requirements:

1) Draw a line as the decision boundary that optimizes the above formulation.
2) Explain in short how you get that line, but you do not need to show a detailed proof.
3) Write a pair of $\mathbf{w}$, b which can define that decision boundary.
4) Circle the support vectors.

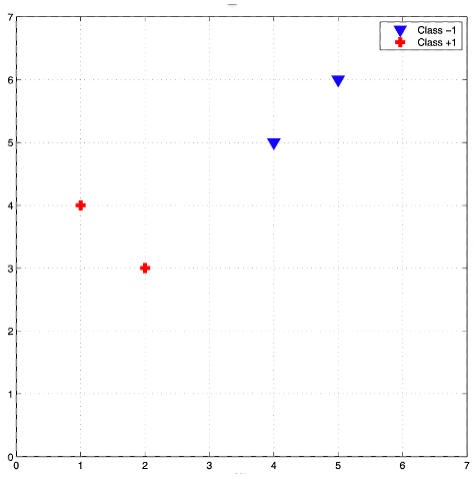

Figure 1: SVM

Problem 2: SVM Experiments(25 points)

In Problem 2 and Problem 3, we utilize the SVC of sklearn library to implement the SVM. The problem formulation of SVC can be found at Section 1.4.7.1. SVC of this link[1]. The SVC function of sklearn at be found at this link[2].

Train and test SVM on the sonar data set which is available at this link[3]. We will use the scaled version for our experiment. A copy of the data is also enclosed in this homework. Use the provided training/testing splitting. In particular, the file "sonar-scale-test-indices.txt" contains the indices of examples in the original file for training, and "sonar-scale-test-indices.txt" contains the indices of examples in the original file for testing. A skeleton code is given in "p2.py".

Requirements

- In all the following requirements, use repeat the experiments for three kernels, "linear", "poly", and "rbf".
- Use the 5-fold cross validation method to decide the best value of the parameter C. The candidate values for C are $0.01, 0.1, 1, 10, 100, 1000$. For each C, report the training accuracy and validation accuracy. Choose the best C that yields the highest validation accuracy.
- Use the selected best C value to train a model on the whole training data, then evaluate and report its performance by accuracy on the testing data.
- Report the results. Compare the results and find which kernel is the best in this case.

Problem 3: Data Preprocessing (40 points)

You are going to use the covtype data set in this question and the next question. This data is described here `https://archive.ics.uci.edu/ml/datasets/Covertype`. The raw data ("covtype.data") is provided in the data folder and a training/testing splitting is also provided (see covtype.train.index and covtype.test.index). Each row in the data file consists of 54 features (the first 54 columns) and the label (the last column). The original data is for a multi-class classification. There are a total of 6 classes. For this problem, you will build a classifier for classifying label "2" (positive) vs the rest (negative). A skeleton code is given in "p3.py".

You are asked to compare different data preprocessing methods. We consider three commonly used data preprocessing methods: rescaling, mean normalization, and standardization. For more details, please read here `https://en.wikipedia.org/wiki/Feature_scaling`. If we let $X \in \mathbb{R}^{n \times d}$ denote the data matrix (n examples and d features), note that the first two methods for conduced for each column and the normalization is conducted for each row. You need to use the same code from the last problem to train a linear SVM classifier (kernel='linear') on the training data with the 5-fold cross-validation to find the best C. You may **NOT** use library to implement the data preprocessing.

Requirements:

[1] `https://scikit-learn.org/stable/modules/svm.html#svm-classification`

[2] `https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html`

[3] `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html#sonar`

- Report in a table the accuracy, F1-score[4], AUC [5] on the testing data for using the raw data and each preprocessed data.
- Plot in a figure the ROC curves for using the raw data and each preprocessed data.
- Discuss your observations of the results.

[4] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html
[5] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.auc.html